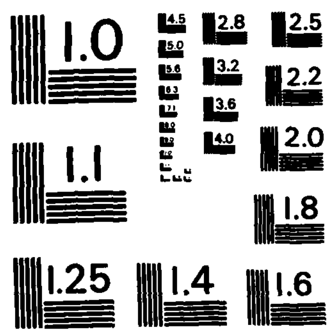
AD-A121 436 TRAM: TARGET REACTION AND MANEUVER - A MONTE CARLO 1/1
SIMULATION(U) NAVAL OCEAN SYSTEMS CENTER SAN DIEGO CA
D H LACKOWSKI APR 82 NOSC-TD-503

UNCLASSIFIED F/G 12/1 NL

END

FILMED

DTIC


1.0
4.5
5.0
5.6
6.3
7.1
8.0
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.1
1.8
1.25
1.4
1.6

C 12

NOSC TD 503

NOSC TD 503

AD A121436

Technical Document 503

TRAM: TARGET REACTION AND MANEUVER — A MONTE CARLO SIMULATION

DH Lackowski

April 1982

Final Report: October-December 1981
Prepared for Naval Sea Systems Command

Approved for public release; distribution unlimited

NOSC

NAVAL OCEAN SYSTEMS CENTER
San Diego, California 92152

NAVAL OCEAN SYSTEMS CENTER, SAN DIEGO, CA 92152

AN ACTIVITY OF THE NAVAL MATERIAL COMMAND

SL GUILLE, CAPT, USN
Commander

HL BLOOD
Technical Director

ADMINISTRATIVE INFORMATION

The work reported herein was sponsored by the Naval Sea Systems Command (NAVSEA 63Y2) under Program Element PE64353N. Work was conducted over the period 1 October–31 December 1981.

Released by
F.E. Rowden, Head
Weapon Systems Office

Under authority of
R.D. Thuleen, Head
Weapon Control and
Sonar Department

CONVERSION TO METRIC

1 ft	≈	0.3 m
1 yard	≈	0.9 m
1 knot	≈	1.8 km/h

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (*When Data Entered*)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER Technical Document 503 (TD 503)	2. GOVT ACCESSION NO. AD-A12143	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (*and Subtitle*) TRAM: Target Reaction and Maneuver–a Monte Carlo Simulation		5. TYPE OF REPORT & PERIOD COVERED Final Report October - December 1981
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) DH Lackowski		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Ocean Systems Center San Diego CA 92152		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 64353N
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Sea Systems Command Washington, DC		12. REPORT DATE April 1982
		13. NUMBER OF PAGES 50
14. MONITORING AGENCY NAME & ADDRESS(*if different from Controlling Office*)		15. SECURITY CLASS. (*of this report*) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (*of this Report*)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (*of the abstract entered in Block 20, if different from Report*)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (*Continue on reverse side if necessary and identify by block number*)

Antisubmarine warfare
Target motion analysis
Weapon delivery accuracy
Hit probability

20. ABSTRACT (*Continue on reverse side if necessary and identify by block number*)

This report describes a computer program which provides a Monte Carlo simulation to determine cumulative hit probability for an ASW weapon delivery system with an evasive submarine as target.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

S/N 0102-LF-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (*When Data Entered*)

Contents

Appendices

1.0 INTRODUCTION

The TRAM program provides a Monte Carlo simulation to determine a cumulative hit probability for an ASW weapon delivery system with an evasive submarine as target. The target evasive maneuver is a course change with a simultaneous speed change order (new thrust level). By user choice, the maneuver parameters may be fixed or may be random within prescribed statistical limits. The program also takes into account random errors of the weapon delivery system. These include errors in target localization, target motion analysis (TMA), and the delivery error of the system.

Hit probability for a given weapon delivery point is determined by a user-provided rectangular grid of hit probabilities. The grid structure provides rectangular areas distributed about the target with a prescribed hit probability associated with weapon delivery in each area.

The program is written in Sperry Univac 1100 Series FORTRAN (ASCII) — Level 9R1. See reference 1. A complete source program listing is contained in Appendix A.

Logical simplicity of the source code was emphasized to make the program and its logical flow more easily understood by the user. In some instances, this resulted in deliberately redundant or inefficient source code.

2.0 MATHEMATICAL DESCRIPTION

Figure 1 presents a representative illustration of target motion and weapon delivery system geometry. The geometry is two-dimensional; target depth is assumed constant.

At time $t = 0$ the target is located at position $T(0)$, located at the origin of the xy coordinate frame, with an initial velocity v_o along the positive x-axis. At some subsequent time $t = T_1$, the target initiates a

---

[1] Sperry Rand Corporation. Sperry Univac Series 1100 FORTRAN (ASCII) - Level 9R1, Programmer Reference. 1979.

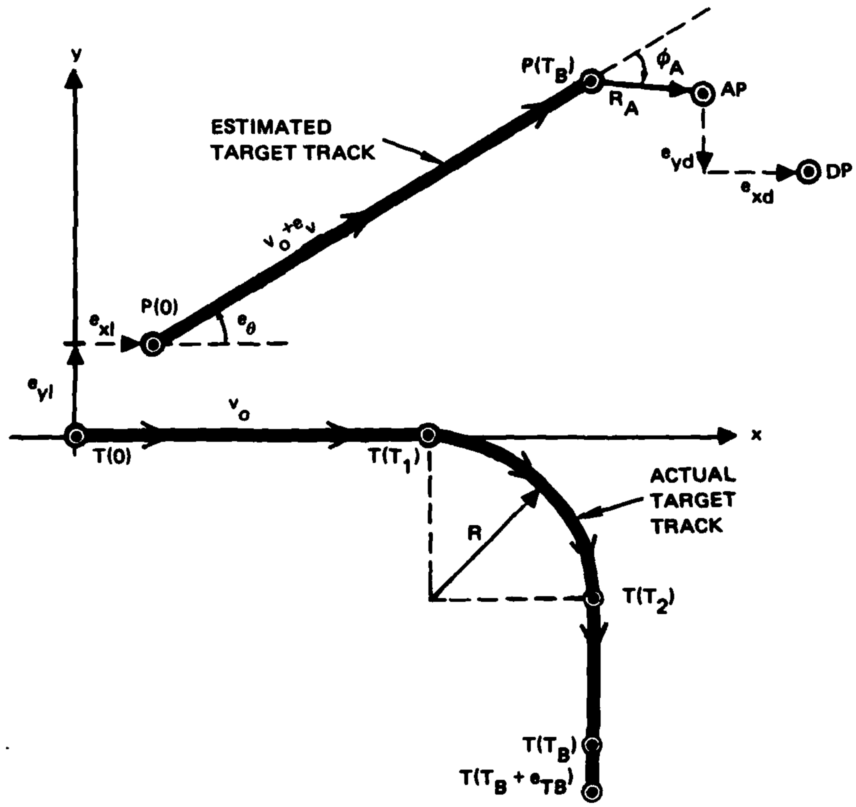


Figure 1. Illustrative run geometry.

maneuver which consists of a course change, or an instantaneous change in thrust level (change in ordered speed), or both, or neither.

The time $t = T_1$ may be prescribed by the user or, otherwise, will be chosen by the program from a random distribution of times uniformly distributed over the interval from $t = 0$ to $t = T_B + e_{TB}$, where T_B is the estimated blind time of the weapon delivery system, ie, the time between last observed target position and weapon delivery, and e_{TB} is the error in the estimate of T_B. e_{TB} is a random variable chosen by the program from a normal distribution, with zero mean and a user-specified standard deviation.

The thrust level change at $t = T_1$ is specified by the user. The course change magnitude may also be specified by the user or, otherwise, is chosen by the program from a uniform distribution between limits provided by the user. The direction of the course change is randomly chosen by the program, with either direction equally probable.

The maneuver event itself may be treated as a random variable by the program. That is, while the user may prescribe that a maneuver does occur, he may, alternatively, provide only a probability that a maneuver occurs. In the latter case, the program will determine the occurrence of the maneuver by comparing a random number, drawn from a uniform distribution over the interval zero to one, with the prescribed probability.

The time $t = T_2$ is the time at which the course change, if any, is completed. Thereafter, the target will proceed on a steady course with constant thrust level until the end of the run at time $t = T_B + e_{TB}$.

Target dynamics during the maneuver are determined from the two-degree-of-freedom equations described in Appendix B.

At time $t = 0$, the weapon delivery system assumes the target to be at position P(0). e_{x1} and e_{y1} are the x and y target localization errors. These errors are chosen by the program from a zero mean normal distribution with a user-specified standard deviation.

The weapon delivery system assumes that the target maintains a steady course e_θ and steady speed $v_o + e_v$ from time $t = 0$ until $t = T_B$. The target course estimate error e_θ is chosen by the program from a zero mean normal distribution with a standard deviation specified by the user. The target speed estimate error e_v is chosen from another zero mean normal distribution with a user specified standard deviation.

At time $t = T_B$, the weapon delivery system's estimated (predicted) target position is at position $P(T_B)$. The aimpoint AP is specified by the user in

terms of range (R_A) and bearing (ϕ_A) offsets relative to the predicted target position at $t = T_B$.

The weapon delivery point is at DP. The delivery point is offset from the aimpoint AP by the weapon delivery errors e_{xd} and e_{yd}. Both are chosen by the program from a zero mean normal distribution with a user-specified standard deviation.

The probability of hit for a given weapon delivery is determined by the location of the weapon delivery point in a user-defined hit probability grid. The rectangular grid is formed by a series of straight lines parallel to the target's course at weapon delivery and by another series of straight lines perpendicular to the target course. An illustrative example is shown on Figure 2.

The target is located at the origin ($x = 0$, $y = 0$) and the target course is along the positive x-axis. The grid lines are specified by the user and may be located anywhere relative to the target. There is no requirement for symmetry about any axis.

The user provides a probability of hit for each rectangle formed by the grid. The left-hand and lower boundaries of each rectangle are assumed to belong to that rectangle. Thus, for example, if the weapon delivery point was located anywhere in the rectangle defined by $500 \leq x < 1000$ yards and $-900 \leq y < -500$ yards, the hit probability for that weapon delivery would be 0.8.

An overall, cumulative hit probability is computed from a set of hit probabilities from individual "runs", where a run is defined as one target trajectory and associated weapon deliveries. A "set" of runs is defined as a collection of runs made under conditions that differ, one run from another, only in a stochastic sense. This structure comprises a Monte Carlo simulation.

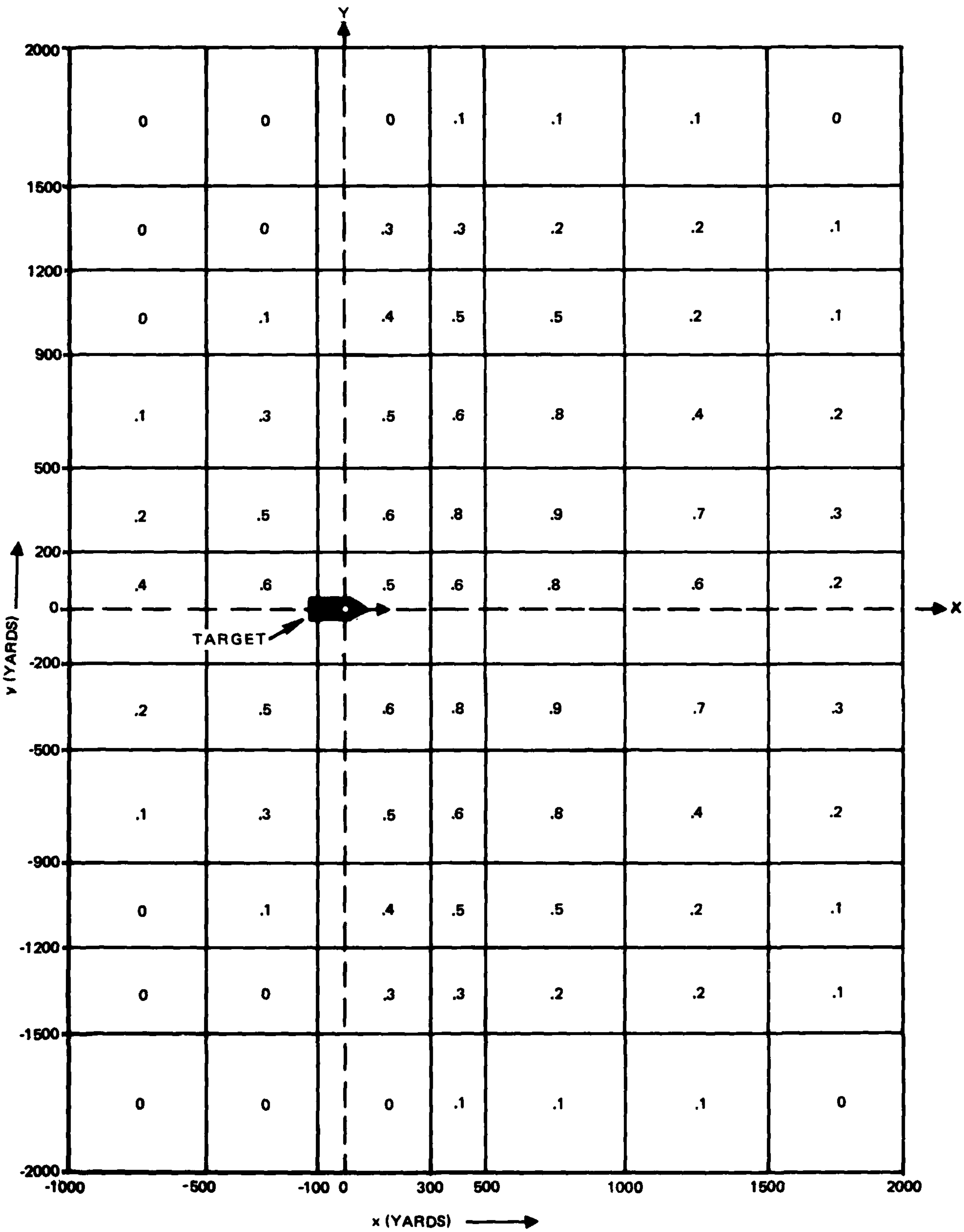


Figure 2. Illustrative hit probability grid.

The user specifies the number of runs in a set and may also specify more than one set of runs for a given program execution. He may also specify any number of weapon deliveries for each run (salvo launch).

For a run with salvo launch of N weapons, the hit probability for the run is computed according to the equation:

$$P_n = P_n' + P_{n-1} - P_n' P_{n-1} \qquad n = 1, 2, \ldots N \tag{1}$$

where P_n is the probability that at least one of the first n weapons will hit and P_n' is the hit probability for the n^{th} weapon.

The cumulative hit probability (Monte Carlo probability) for a set of N runs is computed according to the equation:

$$P = \frac{1}{N} \sum_{k=1}^{N} P_k \tag{2}$$

where P_k is the hit probability for the k^{th} run in the set.

The random variables for each run are provided by two random number generator subroutines: UNFRM and GAUSS. Both generators run continuously (without reset) through a set of runs, so generally different sequences of numbers are provided for different runs in a set. A seed number for the generators is provided by the user at the beginning of each set of runs.

UNFRM generates a sequence of real numbers uniformly distributed over the interval [0,1]. The algorithm used is a multiplicative type of the form:

$$x_{n+1} = (185363)\ x_n \qquad (\mathrm{mod}\ 2^{35}) \tag{3}$$

The interval of the distribution is changed to [0,1] by discarding all $x_k > 2^{27}$ and multiplying the others by 2^{-27}. The 2^{27} factor is a consequence of the integer-to-real conversion process used with the Sperry Univac 1100

Series FORTRAN (ASCII) compiler. See references 1 and 2 for additional details.

The subroutine GAUSS generates a sequence of normally distributed random numbers. The algorithm is based upon the Central Limit theorem and is of the form:

$$x = \left(\sum_{k=1}^{12} y_k - 6 \right) \sigma_x + \bar{x}$$

where x is the normally distributed random variable, the y_k are random numbers uniformly distributed over the interval [0,1] (provided by subroutine UNFRM), σ_x is the standard deviation of the distribution, and $\bar{x}$ is the mean. $\bar{x}$ and σ_x are specified in the program call for the subroutine. See reference 2 for additional details.

Source code listings of UNFRM and GAUSS are included in Appendix A.

3.0 INPUT DATA REQUIREMENTS

For the convenience of the user in formatting input data, all input data variable names are defined with a NAMELIST statement. See reference 1 for details of NAMELIST input.

The program defines three NAMELIST names: NL1, NL2, and NL3. Every input data variable is assigned to one of the three. Upon initiation of program execution, the program searches for and reads-in NL1 data. Upon initiation of each set of runs, the program searches for and reads-in NL2 and NL3 data. For the first set of runs, all NL2 and NL3 data variables must be explicitly defined on an input record (card). For subsequent sets, only those input variables whose values change from the preceding set need to be redefined. If none of the variables in NL2 or NL3 change value from the preceding set, an input card with a blank data field must nevertheless be provided for that

---

[2] Hamming, R. W. Numerical Methods for Scientists and Engineers. McGraw-Hill, 1973.

NAMELIST name, since the program will search for both NL2 and NL3 input cards before executing a set of runs.

Appendix C contains a listing of all input data variable names and a description of the datum each represents. Appendix D contains an example of input data cards to illustrate input data requirements and format.

4.0 SUBROUTINE REQUIREMENTS

The main program calls two subroutines: UNFRM and GAUSS. Subroutine UNFRM (N0, N1, X) returns to the calling program a real number X, which represents one pseudorandom sample from a uniform distribution over the interval [0,1], and an integer number N1, which represents one pseudo-random sample from a set of integers uniformly distributed over the interval $[0,2^{27}]$. Subroutine GAUSS (N0, XMEAN, XSIGMA, N1, X) returns to the calling program a real number X, which represents one pseudorandom sample from a normal distribution with mean XMEAN and standard deviation XSIGMA, and an integer number N1, which represents a pseudorandom sample from a set of integers uniformly distributed over the interval $[0,2^{27}]$.

For both UNFRM and GAUSS, the integer N0 is provided by the calling program. For the first call for either UNFRM or GAUSS, N0 may be any odd integer. If either UNFRM or GAUSS has been called prior to a current call for either subroutine, then N0 should be the value of N1 returned from that call, for either subroutine, which immediately precedes the current call.

Source code listings of UNFRM and GAUSS are included in Appendix A. A mathematical description was provided in Section 2.0. Note that subroutine GAUSS calls the subroutine UNFRM.

5.0 OUTPUT DATA

Appendix E contains the printout for program execution with the input data illustrated in Appendix D.

6.0 TOP LEVEL FLOWCHART

The top-level program flowchart is shown in Figure 3. At the level shown, program execution is essentially controlled by three nested do-loops.

The outermost loop (I=1, NSET) cycles once for each set of runs, while the second loop (J=1, NRUN) cycles once for each run within a set. The innermost loop (K=1, NSHOT) cycles once for each weapon shot of a run. With one exception, lower level flowcharts are not necessary, since program logic is obvious. A second-level flowchart for the one exception, Target Course and Position at Weapon Delivery, is presented in the next section.

7.0 TARGET COURSE AND POSITION AT WEAPON DELIVERY - LEVEL II FLOWCHART

Figure 4 is a level II flowchart for target course and position at weapon delivery. The logical flow of program execution can be discerned from the flowchart and the comments included in the source code listing (Appendix A).

8.0 INTERNAL VARIABLE AND FUNCTION NAMES

Appendix F contains a listing of all internal variable and function names used in the program along with a definition of each.

9.0 REFERENCES

[1] Sperry Rand Corporation. Sperry Univac Series 1100 FORTRAN (ASCII) - Level 9R1, Programmer Reference. 1979.

[2] Hamming, R. W. Numerical Methods for Scientists and Engineers. McGraw-Hill, 1973.

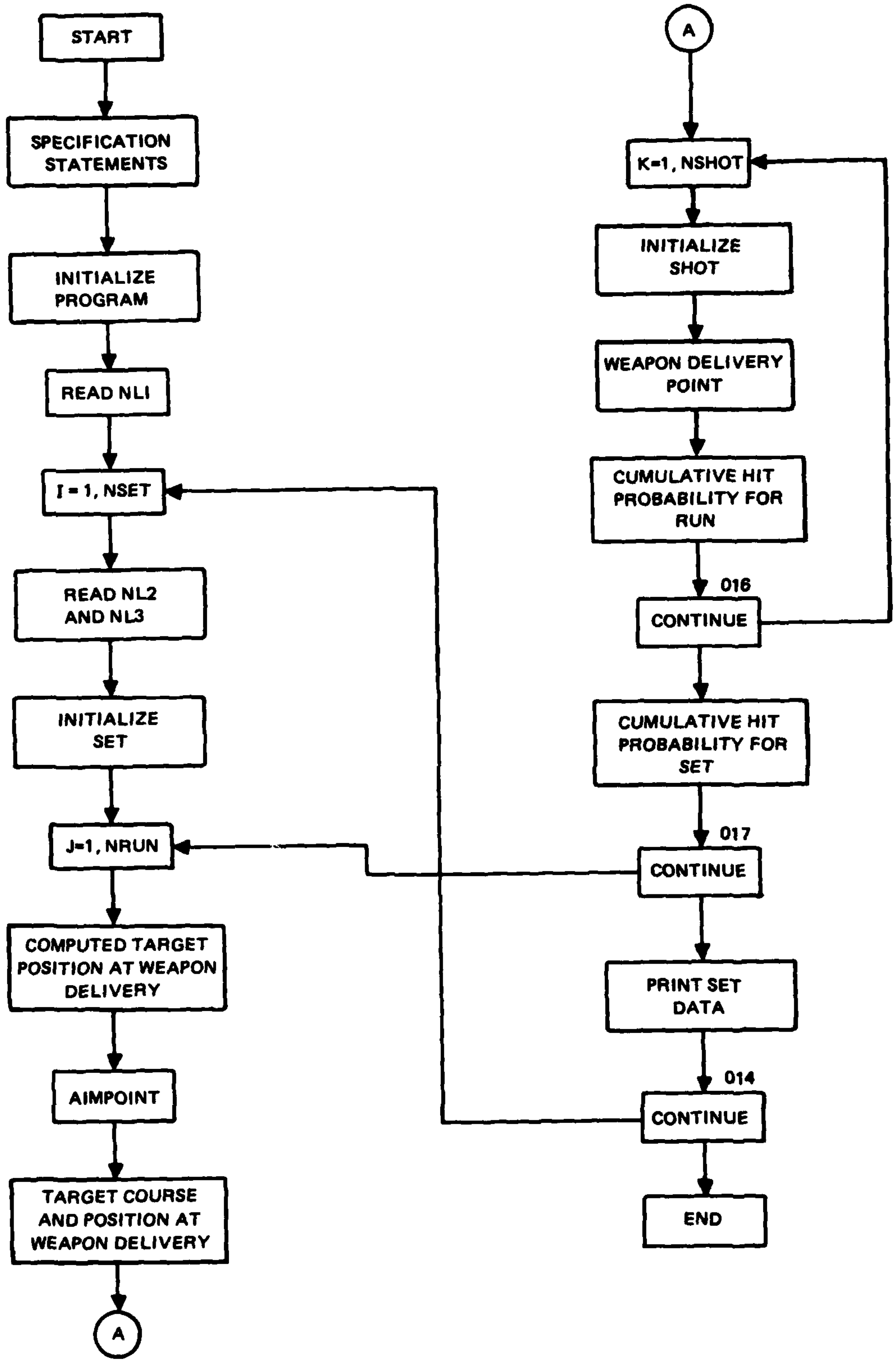


Figure 3. Top-level flow chart

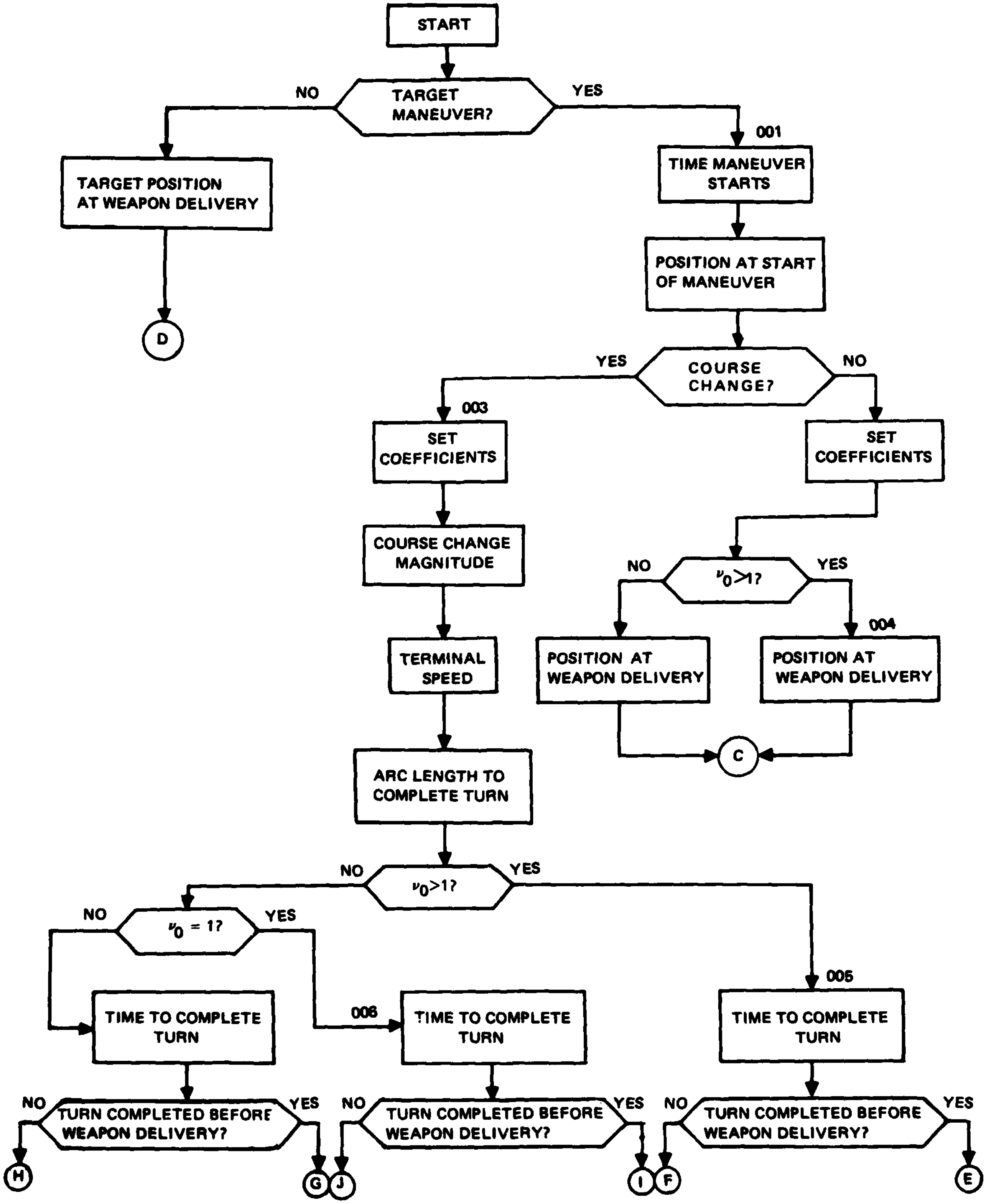


Figure 4. Target course and position at weapon delivery - Level II Flow chart

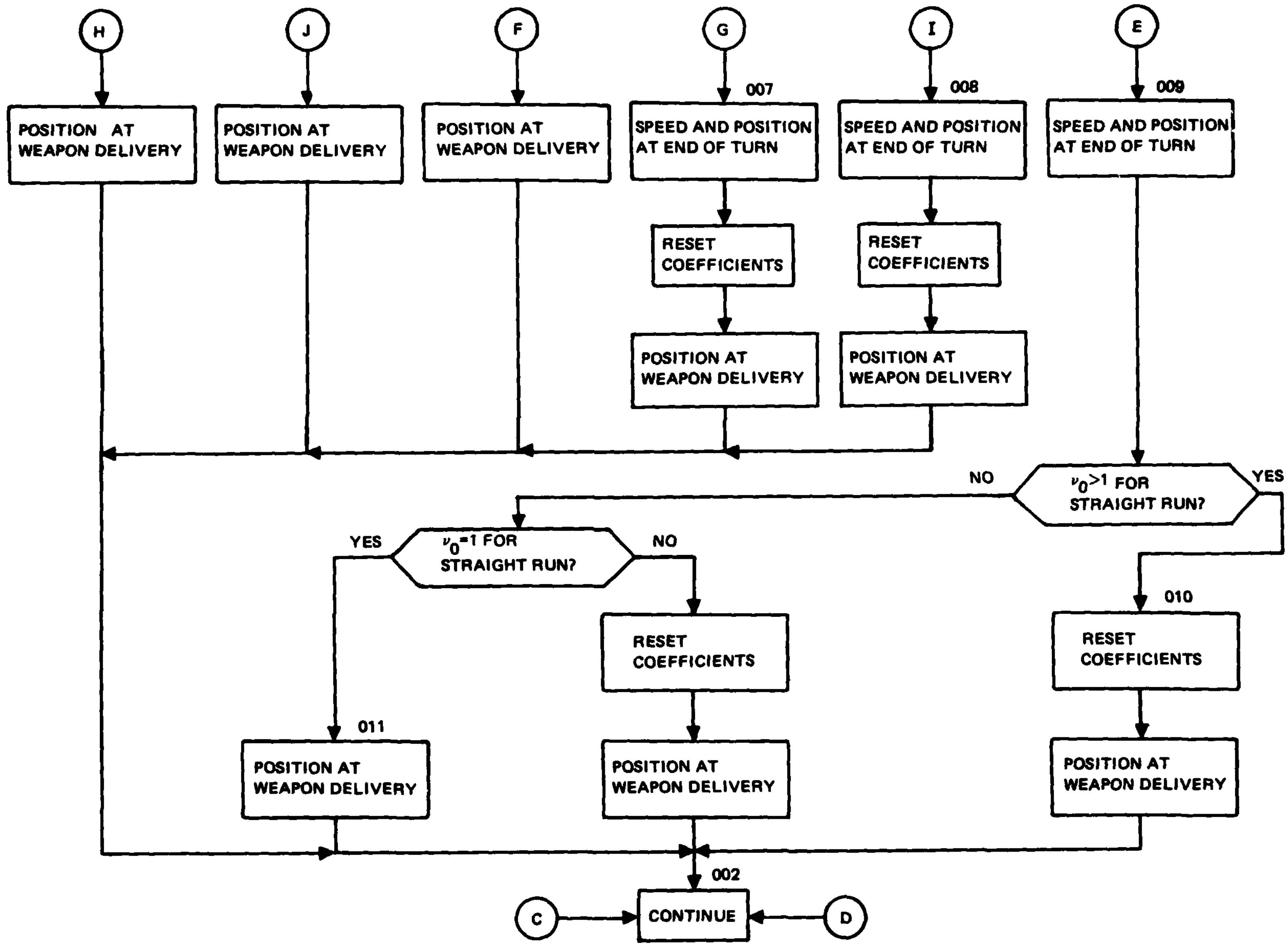


Figure 4. Continued.

APPENDIX A: TRAM SOURCE

CODE LISTING

```
C     THIS PROGRAM COMPUTES A CUMULATIVE HIT PROBABILITY FOR EACH SET OF
C     MONTE-CARLO RUNS FOR AN ASW WEAPON DELIVERY SYSTEM WITH A TARGET
C     THAT MAY EXECUTE A RANDOM MANEUVER DURING THE BLIND TIME INTERVAL
C     BETWEEN TIME OF LAST OBSERVED TARGET POSITION AND TIME OF WEAPON
C     DELIVERY. THE PROGRAM CALLS SUBROUTINES UNFRM AND GAUSS. INPUT
C     DATA ARE THE FOLLOWING:

C        SYMBOL    UNITS    DESCRIPTION

C        NSET      ----     NUMBER OF SETS OF MONTE-CARLO RUNS
C        NRUN      ----     NUMBER OF RUNS IN EACH SET
C        NSHOT     ----     NUMBER OF INDEPENDENT WEAPON DELIVERIES FOR
C                           EACH RUN
C        PTM       ----     PROBABILITY THAT TARGET WILL MANEUVER DURING
C                           BLIND TIME INTERVAL. IF PTM .LT. 0.0 AND PTM
C                           .GT. -1.0, THEN TARGET WILL MANEUVER STARTING
C                           AT -100.0*PTM PERCENT OF BLIND TIME.
C        TS1       KTS      TARGET SPEED BEFORE START OF MANEUVER
C        TS2       KTS      TARGET SPEED ORDERED AT TIME MANEUVER STARTS
C        CMAX      DEG      MAXIMUM MAGNITUDE OF COURSE CHANGE
C        CMIN      DEG      MINIMUM MAGNITUDE OF COURSE CHANGE
C        TR        YDS      TARGET TURN RADIUS
C        TCL       FT       TARGET CHARACTERISTIC LENGTH
C        TR2       YDS**2   TARGET CHARACTERISTIC DRAG RADIUS SQUARED
C        TB        SEC      BLIND TIME
C        APR       YDS      AIMPOINT OFFSET RADIUS
C        APA       DEG      RELATIVE BEARING OF AIMPOINT FROM TARGET
C                           (AIMPOINT OFFSET ANGLE)
C        SIGTB     SEC      STANDARD DEVIATION OF NORMAL ERROR IN ESTIMATE
C                           OF TB
C        SIGTC     DEG      STANDARD DEVIATION OF NORMAL ERROR IN ESTIMATE
C                           OF TARGET INITIAL COURSE
C        SIGTS     KTS      STANDARD DEVIATION OF NORMAL ERROR IN ESTIMATE
C                           OF TARGET INITIAL SPEED
C        SIGL      YDS      STANDARD DEVIATION OF CIRCULAR NORMAL ERROR
C                           IN ESTIMATE OF TARGET INITIAL POSITION
C        SIGD      YDS      STANDARD DEVIATION OF CIRCULAR NORMAL ERROR
C                           OF WEAPON DELIVERY
C        NX        ----     NUMBER OF HIT PROBABILITY GRID LINES
C                           PERPENDICULAR TO TARGET CENTERLINE (25 MAX)
C        NY        ----     NUMBER OF HIT PROBABILITY GRID LINES PARALLEL
C                           TO TARGET CENTERLINE (25 MAX)
C        NRAN      ----     ARBITRARY INTEGER SEED FOR RANDOM NUMBER
C                           GENERATOR
C        GRIDX(I) YDS       ARRAY (25 MAX) OF X COORDINATES OF HIT
C                           PROBABILITY GRID LINES. POSITIVE AHEAD,
C                           NEGATIVE ASTERN. ARRAY IS STRUCTURED IN
C                           ASCENDING ORDER - GRIDX(I) .LT. GRIDX(I+1).
C        GRIDY(I) YDS       ARRAY (25 MAX) OF Y COORDINATES OF HIT
C                           PROBABILITY GRID LINES. POSITIVE TO PORT,
C                           NEGATIVE TO STBD. ARRAY IS STRUCTURED IN
C                           ASCENDING ORDER - GRIDY(I) .LT. GRIDY(I+1).
C        PH(I,J)   ----     ARRAY (24 BY 24 MAX) OF HIT PROBABILITIES.
C                           PH(I,J) IS THE PROBABILITY ASSOCIATED WITH THE
C                           HIT PROBABILITY GRID RECTANGLE DEFINED BY
```

```
C                              ((GRIDX(I) .LE. X) .AND. (GRIDX(I+1) .GT. X))
C                                                    .AND.
C                              ((GRIDY(J) .LE. Y) .AND. (GRIDY(J+1) .GT. Y))

C       FOR THE FIRST SET, ALL INPUT DATA MUST BE PROVIDED. FOR SUBSEQUENT
C       SETS, ONLY DATA MODIFICATIONS ARE REQUIRED. NSET IS READ ONLY ONCE
C       AT THE START OF THE PROGRAM. ALL OTHER DATA ARE READ AT START OF
C       EACH SET. NAMELIST INPUT IS USED.

C       SPECIFICATION STATEMENTS
        DIMENSION GRIDX(25),GRIDY(25),PH(24,24)
        NAMELIST /NL1/NSET
      X           /NL2/NRUN,NSHOT,PTM,TS1,TS2,CMAX,CMIN,TR,TCL,TR2,TB,APR,
      X                 APA,SIGTB,SIGTC,SIGTS,SIGL,SIGD,NX,NY,NRAN
      X           /NL3/GRIDX,GRIDY,PH
    022 FORMAT(1H1,49HCUMULATIVE HIT PROBABILITY FOR ALL RUNS THIS SET:,
      X        2X,6HPCH = ,F7.4//1X,30HINPUT DATA FOR THE SET FOLLOW:)
    026 FORMAT(1H ,14HGRIDX(I) (I=1,,I2,2H):/10(F8.1:2X))
    027 FORMAT(1H ,14HGRIDY(J) (J=1,,I2,2H):/10(F8.1:2X))
    028 FORMAT(1H ,13HPH(I,J) (I=1,,I2,7H),(J=1,,I2,2H):/10(F8.4:2X))

C       DEFINE INVERSE HYPERBOLIC FUNCTIONS
        ASINH(X)=ALOG(X+SQRT(X*X+1.0))
        ACOSH(X)=ALOG(X+SQRT(X*X-1.0))
        ATANH(X)=0.5*ALOG((1.0+X)/(1.0-X))
        ACOTH(X)=0.5*ALOG((X+1.0)/(X-1.0))

C       INITIATE LOOP FOR SETS OF RUNS
        READ(5,NL1)
        DO 014 I=1,NSET

C       READ INPUT DATA FOR SET
        READ(5,NL2)
        READ(5,NL3)

C       INITIALIZE DATA FOR THIS SET
        DUM1=0.562962963
        TS11=DUM1*TS1
        TS21=DUM1*TS2
        SIGTS1=DUM1*SIGTS
        TCL1=TCL/3.0
        DUM1=0.01745329
        APA1=DUM1*APA
        SIGTC1=DUM1*SIGTC
        NRAN1=2*NRAN+1
        PCH=0.0
        FNRUN=FLOAT(NRUN)

C       INITIATE LOOP FOR RUNS IN THIS SET
        DO 017 J=1,NRUN

C       TIME TO WEAPON DELIVERY FOR THIS RUN
        CALL GAUSS(NRAN1,0.0,SIGTB,NRAN1,T1)
        T1=TB+T1

C       COMPUTED TARGET POSITION AT WEAPON DELIVERY
        CALL GAUSS(NRAN1,0.0,SIGL,NRAN1,XTC)
```

```
      CALL GAUSS(NRAN1,0.0,SIGL,NRAN1,YTC)
      CALL GAUSS(NRAN1,0.0,SIGTS1,NRAN1,DUM1)
      DUM1=(TS1I+DUM1)*TB
      CALL GAUSS(NRAN1,0.0,SIGTC1,NRAN1,DUM2)
      XTC=XTC+DUM1*COS(DUM2)
      YTC=YTC-DUM1*SIN(DUM2)

C     AIMPOINT
      DUM2=DUM2+APA1
      XAP=XTC+APR*COS(DUM2)
      YAP=YTC-APR*SIN(DUM2)

C     TARGET COURSE AND POSITION AT WEAPON DELIVERY

C     IS THERE A TARGET MANEUVER
      CALL UNFRM(NRAN1,NRAN1,RN)
      IF(((PTM.LT.0.0).OR.(RN.LT.PTM)).AND.((CMAX.NE.0.0).OR.
     X (TS2.NE.TS1))) GO TO 001

C     TARGET COURSE AND POSITION AT WEAPON DELIVERY. NO MANEUVER.
      TCWD=0.0
      XT=TS1I*T1
      YT=0.0
      GO TO 002

C     TARGET MANEUVER. TIME MANEUVER STARTS.
  001 IF(PTM.LT.0.0) GO TO 012
      CALL UNFRM(NRAN1,NRAN1,RN)
      TM=RN*T1
      GO TO 013
  012 TM=-1.0*PTM*T1
  013 CONTINUE

C     TARGET POSITION AT START OF MANEUVER
      XT=TS1I*TM
      YT=0.0

C     DOES MANEUVER INCLUDE COURSE CHANGE
      CALL UNFRM(NRAN1,NRAN1,RN)
      CCM=CMIN+RN*(CMAX-CMIN)
      IF(CCM.NE.0.0) GO TO 003

C     TARGET COURSE AND POSITION AT WEAPON DELIVERY. SPEED CHANGE ONLY.
      TCWD=0.0
      FNUO=TS1/TS2
      TAU=TS2I*(T1-TM)/TCL1
      IF(FNUO.GT.1.0) GO TO 004

C     TARGET ACCELERATES
      DUM1=ATANH(FNUO)
      XT=XT+TCL1*ALOG(COSH(TAU+DUM1)/COSH(DUM1))
      GO TO 002

C     TARGET DECELERATES
  004 DUM1=ACOTH(FNUO)
      XT=XT+TCL1*ALOG(SINH(TAU+DUM1)/SINH(DUM1))
      GO TO 002
```

```
C      TARGET COURSE AND POSITION AT WEAPON DELIVERY. COURSE CHANGE.
   003 DUM1=1.0+TR2/(TR*TR)
       CR=TCLI/DUM1

C      MAGNITUDE AND DIRECTION OF COURSE CHANGE
       CCM=0.01745329*CCM
       CALL UNFRN(NRAN1,NRAN1,RN)
       CCS=1.0
       IF(RN.GT.0.5) CCS=-1.0

C      TERMINAL SPEED FOR INFINITE TURN
       TST=TS2I/SQRT(DUM1)

C      ARC LENGTH OF FULL TURN
       S=TR*CCM

C      DOES TARGET ACCELERATE OR DECELERATE DURING TURN
       FNUO=TS1I/TST
       IF(FNUO.GT.1.0) GO TO 005
       IF(FNUO.EQ.1.0) GO TO 006

C      TIME TO COMPLETE TURN. TARGET ACCELERATES.
       DUM1=ATANH(FNUO)
       T2=(CR/TST)*(ACOSH(EXP(S/CR)*COSH(DUM1))-DUM1)

C      CAN TURN BE COMPLETED BEFORE WEAPON DELIVERY
       IF((TM+T2).LE.T1) GO TO 007

C      COURSE AND POSITION DURING TURN AT WEAPON DELIVERY
       TAU=(TST/CR)*(T1-TM)
       S=CR*ALOG(COSH(TAU+DUM1)/COSH(DUM1))
       DUM2=S/TR
       TCWD=DUM2*CCS
       XT=XT+TR*SIN(DUM2)
       YT=TR*(1.0-COS(DUM2))*CCS
       GO TO 002

C      SPEED AND POSITION AT END OF TURN
   007 XT=XT+TR*SIN(CCM)
       YT=TR*(1.0-COS(CCM))*CCS
       TAU=(TST/CR)*T2
       TSE=TST*TANH(TAU+DUM1)

C      RESET COEFFICIENTS FOR STRAIGHT RUN AFTER TURN
       FNUO=TSE/TS2I
       TAU=(TS2I/CR)*(T1-TM-T2)
       DUM1=ATANH(FNUO)

C      COURSE AND POSITION AT WEAPON DELIVERY
       TCWD=CCM*CCS
       DUM2=TCLI*ALOG(COSH(TAU+DUM1)/COSH(DUM1))
       XT=XT+DUM2*COS(CCM)
       YT=YT+DUM2*SIN(CCM)*CCS
       GO TO 002

C      TIME TO COMPLETE TURN. TARGET SPEED CONSTANT.
```

```
  006 T2=TR*CCM/TS1I

C     CAN TURN BE COMPLETED BEFORE WEAPON DELIVERY
      IF((TM+T2).LE.TI) GO TO 008

C     COURSE AND POSITION DURING TURN AT WEAPON DELIVERY
      DUM2=CCM*(TI-TM)/T2
      TCWD=DUM2*CCS
      XT=XT+TR*SIN(DUM2)
      YT=TR*(1.0-COS(DUM2))*CCS
      GO TO 002

C     POSITION AT END OF TURN
  008 XT=XT+TR*SIN(CCM)
      YT=TR*(1.0-COS(CCM))*CCS

C     RESET COEFFICIENTS FOR STRAIGHT RUN AFTER TURN
      FNUO=TS1I/TS2I
      TAU=(TS2I/CR)*(TI-TM-T2)
      DUM1=ATANH(FNUO)

C     COURSE AND POSITION AT WEAPON DELIVERY
      TCWD=CCM*CCS
      DUM2=TCLI*ALOG(COSH(TAU+DUM1)/COSH(DUM1))
      XT=XT+DUM2*COS(CCM)
      YT=YT+DUM2*SIN(CCM)*CCS
      GO TO 002

C     TIME TO COMPLETE TURN. TARGET DECELERATES
  005 DUM1=ACOTH(FNUO)
      T2=(CR/TST)*(ASINH(EXP(S/CR)*SINH(DUM1))-DUM1)

C     CAN TURN BE COMPLETED BEFORE WEAPON DELIVERY
      IF((TM+T2).LE.TI) GO TO 009

C     COURSE AND POSITION DURING TURN AT WEAPON DELIVERY
      TAU=(TST/CR)*(TI-TM)
      S=CR*ALOG(SINH(TAU+DUM1)/SINH(DUM1))
      DUM2=S/TR
      TCWD=DUM2*CCS
      XT=XT+TR*SIN(DUM2)
      YT=TR*(1.0-COS(DUM2))*CCS
      GO TO 002

C     SPEED AND POSITION AT END OF TURN
  009 XT=XT+TR*SIN(CCM)
      YT=TR*(1.0-COS(CCM))*CCS
      TAU=(TST/CR)*T2
      TSE=TST/TANH(TAU+DUM1)

C     DOES TARGET ACCELERATE OR DECELERATE AFTER TURN
      FNUO=TSE/TS2I
      IF(FNUO.GT.1.0) GO TO 010
      IF(FNUO.EQ.1.0) GO TO 011

C     TARGET ACCELERATES AFTER TURN. RESET COEFFICIENTS FOR STRAIGHT
C     RUN.
```

```
      TAU=(TS2I/CR)*(T1-TM-T2)
      DUM1=ATANH(FNUO)

C     COURSE AND POSITION AT WEAPON DELIVERY
      TCWD=CCM*CCS
      DUM2=TCL1*ALOG(COSH(TAU+DUM1)/COSH(DUM1))
      XT=XT+DUM2*COS(CCM)
      YT=YT+DUM2*SIN(CCM)*CCS
      GO TO 002

C     TARGET SPEED CONSTANT AFTER TURN, COURSE AND POSITION AT WEAPON
C     DELIVERY.
  011 TCWD=CCM*CCS
      DUM2=TSE*(T1-TM-T2)
      XT=XT+DUM2*COS(CCM)
      YT=YT+DUM2*SIN(CCM)*CCS
      GO TO 002

C     TARGET DECELERATES AFTER TURN, RESET COEFFICIENTS FOR STRAIGHT
C     RUN.
  010 TAU=(TS2I/CR)*(T1-TM-T2)
      DUM1=ACOTH(FNUO)

C     COURSE AND POSITION AT WEAPON DELIVERY
      TCWD=CCM*CCS
      DUM2=TCL1*ALOG(SINH(TAU+DUM1)/SINH(DUM1))
      XT=XT+DUM2*COS(CCM)
      YT=YT+DUM2*SIN(CCM)*CCS

C     TERMINAL FOR TARGET COURSE AND POSITION
  002 CONTINUE

C     INITIALIZE LOOP FOR NUMBER OF SHOTS ON RUN
      PCHS=0.0
      DO 016 K=1,NSHOT

C     WEAPON DELIVERY POINT
      CALL GAUSS(NRAN1,0.0,SIGD,NRAN1,XWD)
      CALL GAUSS(NRAN1,0.0,SIGD,NRAN1,YWD)
      XWD=XWD+XAP
      YWD=YWD+YAP

C     CONVERT WEAPON DELIVERY POINT TO HIT PROBABILITY GRID COORDINATES
      DUM1=SIN(TCWD)
      DUM2=COS(TCWD)
      DUM3=XWD-XT
      DUM4=YWD-YT
      XWDG=DUM4*DUM1+DUM3*DUM2
      YWDG=DUM4*DUM2-DUM3*DUM1

C     IS THE WEAPON DELIVERY POINT OUTSIDE THE HIT PROBABILITY GRID
      IF((XWDG.LT.GRIDX(1)).OR.(XWDG.GE.GRIDX(NX)).OR.
     X   (YWDG.LT.GRIDY(1)).OR.(YWDG.GE.GRIDY(NY))) GO TO 016

C     CUMULATIVE HIT PROBABILITY FOR RUN
      II=1
  020 IF((GRIDX(II).LE.XWDG).AND.(GRIDX(II+1).GT.XWDG)) GO TO 019
```

```
      II=II+1
      GO TO 020
  019 JJ=1
  018 IF((GRIDY(JJ).LE.YWDG).AND.(GRIDY(JJ+1).GT.YWDG)) GO TO 021
      JJ=JJ+1
      GO TO 018
  021 PCHS=PCHS+PH(II,JJ)-PCHS*PH(II,JJ)
C     LOOP TERMINAL FOR NUMBER OF SHOTS ON THIS RUN
  016 CONTINUE
C     CUMULATIVE HIT PROBABILITY FOR SET
      PCH=PCH+PCHS/FNRUN
C     LOOP TERMINAL FOR RUNS IN THIS SET
  017 CONTINUE
C     PRINT RESULTS FOR SET
      WRITE(6,22) PCH
      WRITE(6,NL2)
      WRITE(6,26) NX,(GRIDX(II),II=1,NX)
      WRITE(6,27) NY,(GRIDY(II),II=1,NY)
      NXI=NX-1
      NYI=NY-1
      WRITE(6,28) NXI,NYI,((PH(II,JJ),II=1,NXI),JJ=1,NYI)
C     LOOP TERMINAL FOR SETS OF RUNS
  014 CONTINUE
      END
```

```
C      SUBROUTINE UNFRM(NO,N1,X) COMPUTES A REAL PSEUDO-RANDOM VARIABLE X
C      UNIFORMLY DISTRIBUTED OVER THE INTERVAL 0.0 TO 1.0 AND AN INTEGER
C      PSEUDO-RANDOM VARIABLE N1 UNIFORMLY DISTRIBUTED OVER THE INTERVAL
C      0 TO (2**35-1)-(2**8-1). IF A CALL FOR UNFRM HAS NOT BEEN
C      PRECEEDED BY ANOTHER CALL FOR UNFRM OR A CALL FOR GAUSS, THE SEED
C      VARIABLE NO MUST BE AN ODD INTEGER. OTHERWISE NO SHOULD BE THE
C      VALUE OF N1 RETURNED BY THE PRECEEDING CALL FOR UNFRM OR GAUSS.
C      THIS SUBROUTINE IS INTENDED FOR USE WITH A COMPUTER HAVING AN
C      INTEGER WORD LENGTH OF 35 BITS PLUS SIGN AND AN INTEGER-TO-REAL
C      CONVERSION FUNCTION FLOAT(I) WHICH RETAINS ONLY THE 27 MS BITS
C      OF THE INTEGER. THE PERIOD OF THE PSEUDO RANDOM NUMBERS IS
C      APPROXIMATELY 2**33.
       SUBROUTINE UNFRM(NO,N1,X)
   002 N1=185363*NO
       IF(N1.GE.0) GO TO 001
       N1=-1*N1
   001 IF(N1.GT.34359738112) GO TO 002
       X=FLOAT(N1)/34359738112.0
       RETURN
       END
```

```
C      SUBROUTINE GAUSS(NO,XM,XS,NI,X) COMPUTES A REAL, NORMALLY
C      DISTRIBUTED, PSEUDO-RANDOM VARIABLE X WITH MEAN XM AND STANDARD
C      DEVIATION XS. IF A CALL FOR GAUSS HAS NOT BEEN PRECEEDED BY
C      ANOTHER CALL FOR GAUSS OR A CALL FOR UNFRM,THE SEED VARIABLE NO
C      MUST BE AN ODD INTEGER, OTHERWISE NO SHOULD BE THE VALUE OF NI
C      RETURNED BY THE PRECEEDING CALL FOR GAUSS OR UNFRM. SUBROUTINE
C      GAUSS CALLS THE SUBROUTINE UNFRM.
       SUBROUTINE GAUSS(NO,XM,XS,NI,X)
       X=0.0
       DO 001 I=1,12
       CALL UNFRM(NO,NO,DX)
       X=X+DX
   001 CONTINUE
       NI=NO
       X=(X-6.0)*XS+XM
       RETURN
       END
```

APPENDIX B: TWO-DEGREE-OF-FREEDOM EQUATIONS OF MOTION FOR A MANEUVERING TARGET

TERMINAL SPEED FOR A GIVEN THRUST LEVEL

The terminal (steady-state) speed v of a submerged body with constant thrust level T is given by:

$$v = \sqrt{\frac{2T}{\rho C_D S}} \tag{B-1}$$

where:

ρ = density of the medium

C_D = drag coefficient

S = characteristic area

Now let C_{D1} denote the drag coefficient of the body for motion along a straight path and C_{D2} the drag coefficient of the body in a turn. Let v_1 denote the terminal speed of the body for a straight run with constant thrust T and v_2 the terminal speed in an infinite turn with the same thrust level T. Then, it follows from equation (B-1):

$$v_2 = v_1 \sqrt{\frac{C_{D1}}{C_{D2}}} \tag{B-2}$$

Assume C_{D2} is of the form:

$$C_{D2} = C_{D1} + KC_L{}^2 \tag{B-3}$$

where K is some constant and C_L is the lift coefficient due to turning. Then:

$$\frac{C_{D1}}{C_{D2}} = \left(1 + \frac{KC_L^2}{C_{D1}}\right)^{-1} \tag{B-4}$$

The lift L for a body at speed v during a turn of constant radius R is given by:

$$L = \frac{1}{2}\rho v^2 C_L S = \frac{2mv^2}{R} \qquad \text{(B-5)}$$

where m is the body mass and is equal to the mass of the water displaced by the body (we assume trim for neutral buoyancy).

Then, solving for C_L:

$$C_L = \frac{4m}{\rho SR} \qquad \text{(B-6)}$$

Substituting (B-6) into (B-4):

$$\frac{C_{D1}}{C_{D2}} = \left(1 + \frac{16\ Km^2}{\rho^2 S^2 R^2 C_{D1}}\right)^{-1} \qquad \text{(B-7)}$$

Or

$$\frac{C_{D1}}{C_{D2}} = \left(1 + \frac{R_c^2}{R^2}\right)^{-1} \qquad \text{(B-8)}$$

where

$$R_c^2 = \frac{16\ Km^2}{\rho^2 S^2\ C_{D1}} \qquad \text{(B-9)}$$

R_c^2 is defined as the characteristic drag radius squared of the body. Then, from equations (B-2) and (B-8):

$$v_2 = v_1 \left(1 + \frac{R_c^2}{R^2}\right)^{-\frac{1}{2}} \qquad \text{(B-10)}$$

Equation (B-10) is used to compute the terminal speed v_2 for a target in a circular turn of constant radius R, at a constant thrust level which would produce a terminal speed v_1 if the target were on a straight run ($R = \infty$). This formulation is a consequence of the fact that thrust levels for a ship or submarine are ordered in terms of the corresponding straight-run terminal speed.

TARGET MANEUVERS

The program provides for a target maneuver which consists of an instantaneous thrust change and/or a circular turn maneuver with the start of the turn at the time of thrust change.

We define the following:

v = instantaneous target speed

v_o = initial target speed at time maneuver starts (t=0)

T_o = thrust level before start of maneuver

T_T = thrust level after start of maneuver

v_T = terminal speed for the maneuver at thrust level T_T

During the maneuver, the force acting on the target is:

$$F = T_T - \frac{1}{2}\rho v^2 C_{D2} S \qquad \text{(B-11)}$$

$$= \frac{1}{2}\rho C_{D2} S \left(v_T^2 - v^2\right)$$

Then the target acceleration is:

$$a = c\left(v_T^2 - v^2\right) \qquad \text{(B-12)}$$

where:

$$c = \frac{\rho C_{D2} S}{2m} \tag{B-13}$$

From equation (B-8):

$$c = \frac{\rho C_{D1} S}{2m}\left(1 + \frac{R_c^2}{R^2}\right) \tag{B-14}$$

$$= \frac{1}{\ell_c}\left(1 + \frac{R_c^2}{R^2}\right)$$

The term ℓ_c is called the characteristic length of the target:

$$\ell_c = \frac{2m}{\rho C_{D1} S} \tag{B-15}$$

From equation (B-12) we obtain the differential equation:

$$\frac{d\nu}{1-\nu^2} = cv_T dt = K dt \tag{B-16}$$

where

$$\nu = \frac{v}{v_T} \tag{B-17}$$

and

$$K = cv_T \tag{B-18}$$

The solution of the differential equation (B-16) by integration depends upon the value of ν. We identify three cases:

Case I: Target accelerates during maneuver ($\nu < 1$)

Case II: Target decelerates during maneuver ($\nu > 1$)

Case III: Target speed constant during maneuver ($\nu = 1$)

<u>Case I: Target Accelerates during Maneuver ($\nu < 1$)</u>

For $\nu < 1$, the solution of the differential equation (B-16) is:

$$\tanh^{-1}\nu - \tanh^{-1}\nu_o = Kt \tag{B-19}$$

where:

$$\nu_o = \frac{v_o}{v_T} \tag{B-20}$$

Or:

$$\nu = \tanh\left(Kt + \tanh^{-1}\nu_o\right) \tag{B-21}$$

From equations (B-17) and (B-18) we see that

$$\nu = \frac{cv}{K} \tag{B-22}$$

Or

$$\nu = \left(\frac{c}{K}\right) \frac{ds}{dt} \tag{B-23}$$

where s is the path length variable during the maneuver.

Then equation (B-21) may be rewritten:

$$c\ ds = K \tanh\left(Kt + \tanh^{-1}\nu_o\right) dt \tag{B-24}$$

Integration of equation (B-24) yields:

$$s = \frac{1}{c} \ln\left[\frac{\cosh\left(Kt + \tanh^{-1}\nu_o\right)}{\cosh\left(\tanh^{-1}\nu_o\right)}\right] \tag{B-25}$$

Equation (B-25) defines the path length traveled as a function of time. In the case of a turn maneuver, this is equivalent to the angle of turn (course change), since the turn is assumed to be circular (constant radius).

To compute the time required to travel a given path length, or, in the case of a turn, to complete a given course change, we solve equation (B-25) for t:

$$t = \frac{1}{K}\left\{\cosh^{-1}\left[e^{cs}\cosh(\tanh^{-1}\nu_o)\right] - \tanh^{-1}\nu_o\right\} \tag{B-26}$$

Target speed at any time during the maneuver is computed with equation (B-21).

<u>Case II: Target Accelerates during Maneuver ($\nu > 1$)</u>

For $\nu > 1$, the differential equation (B-16) has a solution:

$$\coth^{-1}\nu - \coth^{-1}\nu_o = Kt \tag{B-27}$$

If we proceed from equation (B-27) in the same manner as we did from equation (B-19) for Case I ($\nu < 1$), we obtain:

$$\nu = \coth\left(Kt + \coth^{-1}\nu_o\right) \tag{B-28}$$

$$s = \frac{1}{c}\,\ln\left[\frac{\sinh\left(Kt + \coth^{-1}\nu_o\right)}{\sinh\left(\coth^{-1}\nu_o\right)}\right] \tag{B-29}$$

$$t = \frac{1}{K}\left\{\sinh^{-1}\left[e^{cs}\sinh\left(\coth^{-1}\nu_o\right)\right] - \coth^{-1}\nu_o\right\} \tag{B-30}$$

Equation (B-28) provides target speed at any time during the maneuver, while (B-29) provides distance traveled (or course change) since start of the maneuver. Equation (B-30) provides the time required to travel a given distance, or, in the case of a turn, the time required to complete a given course change.

Case III: Target Speed Constant During Maneuver ($\nu = 1$)

For the case of constant speed during the maneuver ($\nu = 1$), the equations of motion are trivial, since we have linear motion at constant speed or motion along a circular arc at constant speed.

APPENDIX C: INPUT DATA VARIABLES

NAMELIST NL1

NSET:	Number of sets of runs

NAMELIST NL2

NRUN:	Number of runs in the set
NSHOT:	Number of weapons simultaneously launched for each run.
PTM:	Probability that the target will initiate a maneuver (course and/or thrust change) at some time during the blind time interval. If $-1.0 < PTM < 0.0$, then the target will initiate a maneuver at $-100.0 * PTM$ percent of the blind time.
TS1:	Initial target speed (knots).
TS2:	New target speed ordered at time maneuver is initiated (knots).
CMAX:	Maximum magnitude of target course change (degrees).
CMIN:	Minimum magnitude of target course change (degrees).
TR:	Target turn radius for course change (yards)
TCL:	Target characteristic length (feet)
TR2:	Target characteristic drag radius squared (yards squared)
TB:	Estimated blind time (seconds)

APR: Aimpoint offset radius. Distance of aimpoint from estimated target position at estimated time of weapon delivery (yards).

APA: Aimpoint offset angle. Relative bearing of aimpoint from estimated target position and course at estimated time of weapon delivery. (degrees)

SIGTB: Standard deviation of normal error in estimate of blind time. (seconds)

SIGTC: Standard deviation of normal error in estimate of target course (degrees)

SIGTS: Standard deviation of normal error in estimate of target speed (knots)

SIGL: Standard deviation of normal target localization errors (yards).

SIGD: Standard deviation of normal weapon delivery errors (yards).

NX: Number of hit probability grid lines perpendicular to target centerline (program limit: 25 maximum)

NY: Number of hit probability grid lines parallel to target centerline (program limit: 25 maximum)

NRAN: Seed for random number generators. May be any integer less than 2^{35} in magnitude.

NAMELIST NL3

GRIDX(I), (I=1, NX): Distance from target, along target centerline, to I^{th} hit probability grid line perpendicular to centerline. Positive ahead, negative astern. Array must be ordered such that GRIDX(I) < GRIDX(I+1).

GRIDY(I), (I=1, NY): Distance from target centerline to Ith hit probability grid line parallel to centerline. Positive to port, negative to starboard. Array must be ordered such that GRIDY(I) < GRIDY(I+1).

PH(I,J), (I=1, NX-1), (J=1, NY-1): Probability of hit for hit probability grid rectangle defined by GRIDX(I) $\leq$ x < GRIDX(I+1) and GRIDY(J) $\leq$ y < GRIDY (J+1)

APPENDIX D: INPUT DATA CARDS

The following key punch work sheets represent a typical set of input data cards for program execution, with each line of each work sheet representing one card. The numbers shown in columns 79-80 are not part of the input data and would not be included in the actual input cards. The numbers are included only for reference purposes in the text of this appendix.

Card 01 specifies that program execution will consist of four sets of runs. Parameters for the initial set of runs are contained in cards 02-21. The hit probability grid parameters, contained on cards 07-21, correspond to the hit probability grid shown on Figure (2).

Input parameters for the second set of runs are provided by cards 22-23. These specify that input parameters for the second set of runs will be identical to these of the first, except for a different seed (NRAN) for the random number generators. Thus sets one and two can test for stochastic convergence by comparing results for generally different sequences of random variables from the same statistical distributions. Note that, even though there are no changes to NL3 namelist data for the second set, a blank NL3 card (#23) is provided, as required, for proper program execution.

Card 24 changes the magnitude of target turn maneuvers and the random number generator seed for the third set. Card 25 provides the required card for NL3, even though there are no NL3 data changes.

Cards 26-27 specify that the input parameters for the fourth set will be identical to those of the third, except for the random number generator seed.

KEY PUNCH WORK SHEET
1190 GEN-10462 1 5 71

PAGE OF

```
$NL1  NSET=4 $END                                                              01
$NL2  NRUN=4ØØØ,NSHOT=1,                                                       02
      PTM=Ø.5,TS1=1Ø,TS2=25,CMAX=6Ø,CMIN=3Ø,                                   03
      TR=15Ø,TCL=168Ø,TR2=387ØØ,                                               04
      TB=12Ø,APR=75Ø,APA=3Ø,                                                   05
      SIGTB=2,SIGTC=1,SIGTS=Ø.5,SIGL=1ØØ,SIGD=5ØØ,                             06
      NX=8,NY=12,NRAN=11482 $END                                               07
$NL3  GRIDX(1)=-1ØØØ,-5ØØ,-1ØØ,3ØØ,5ØØ,1ØØØ,15ØØ,2ØØØ,                         08
      GRIDY(1)=-2ØØØ,-15ØØ,-12ØØ,-9ØØ,-5ØØ,-2ØØ,2ØØ,5ØØ,9ØØ,12ØØ,15ØØ,          09
                  2ØØØ,                                                        10
      PH(1,1)=3*Ø,3*Ø.1,Ø,                                                     11
      PH(1,2)=2*Ø,2*Ø.3,2*Ø.2,Ø.1,                                             12
      PH(1,3)=Ø,Ø.1,Ø.4,2*Ø.5,Ø.2,Ø.1,                                         13
      PH(1,4)=Ø.1,Ø.3,Ø.5,Ø.6,Ø.8,Ø.4,Ø.2,                                     14
      PH(1,5)=Ø.2,Ø.5,Ø.6,Ø.8,Ø.9,Ø.7,Ø.3,                                     15
      PH(1,6)=Ø.4,Ø.6,Ø-5,Ø.6,Ø.8,Ø.6,Ø.2,                                     16
      PH(1,7)=Ø.2,Ø.5,Ø.6,Ø.8,Ø.9,Ø.7,Ø.3,                                     17
      PH(1,8)=Ø.1,Ø.3,Ø.5,Ø.6,Ø.8,Ø.4,Ø.2,                                     18
      PH(1,9)=Ø,Ø.1,Ø.4,2*Ø.5,Ø.2,Ø.1,                                         19
      PH(1,10)=2*Ø,2*Ø.3,2*Ø.2,Ø.1,                                            20
```

KEY PUNCH WORK SHEET
1WD-GEN-10462 1 (5-71)

NAME | EXT | PAGE OF

```
      PN(1,11)=3*Ø,3*Ø.1,Ø $END                                                21
 $NL2 NRAN=3382 $END                                                           22
 $NL3 $END                                                                     23
 $NL2 NRAN=11482,CMAX=9Ø,CMIN=6Ø $END                                          24
 4NL3 $END                                                                     25
 $NL2 NRAN=3382 $END                                                           26
 $NL3 $END                                                                     27
```

APPENDIX E: PROGRAM OUTPUT DATA AND FORMAT

```
CUMULATIVE HIT PROBABILITY FOR ALL RUNS THIS SET:  PCH =   .4075

INPUT DATA FOR THE SET FOLLOW:
 $NL2
NRUN =        4000,NSHOT =          1,PTM = .50000000+000,TS1 = .10000000+002,TS2 = .25000000+002,CMAX = .60000000+002,
CMIN = .30000000+002,TR = .15000000+003,TCL = .16800000+004,TR2 = .38700000+005,TB = .12000000+003,APR = .75000000+003,
APA = .30000000+002,SIGTR = .20000000+001,SIGTC = .10000000+001,SIGTS = .50000000+000,SIGL = .10000000+003,SIGD = .50000000+003,
NX =          8,NY =          12,NRAN =        11482
$END
GRIDX(I) (I=1, 8):
-1000.0     -500.0     -100.0      300.0      500.0     1000.0     1500.0     2000.0
GRIDY(J) (J=1,12):
-2000.0    -1500.0    -1200.0     -900.0     -500.0     -200.0      200.0      500.0      900.0     1200.0
 1500.0     2000.0
PH(I,J) (I=1, 7),(J=1,11):
  .0000      .0000      .0000      .1000      .1000      .1000      .0000      .0000      .0000      .3000
  .3000      .2000      .2000      .1000      .0000      .1000      .4000      .5000      .5000      .2000
  .1000      .1000      .3000      .5000      .6000      .8000      .4000      .2000      .2000      .5000
  .6000      .8000      .9000      .7000      .3000      .4000      .6000      .5000      .6000      .8000
  .6000      .2000      .2000      .5000      .4000      .8000      .9000      .7000      .3000      .1000
  .3000      .5000      .6000      .8000      .4000      .2000      .0000      .1000      .4000      .5000
  .5000      .7000      .1000      .0000      .0000      .3000      .3000      .2000      .2000      .1000
  .0000      .0000      .0000      .1000      .1000      .1000      .0000
```

```
CUMULATIVE HIT PROBABILITY FOR ALL RUNS THIS SET:  PCH =   .4067

INPUT DATA FOR THE SET FOLLOW:
 $NL2
NRUN =       4000,NSHOT =           1,PTH = .50000000+000,TS1 = .10000000+002,TS2 = .25000000+002,CMAX = .60000000+002,
CMIN = .30000000+002,TR = .15000000+003,TCL = .16800000+004,TR2 = .38700000+005,TB = .12000000+003,APR = .75000000+003,
APA = .30000000+002,SIGTB = .20000000+001,SIGTC = .10000000+001,SIGTS = .50000000+000,SIGL = .10000000+003,SIGD = .50000000+003,
NX =          8,NY =          12,NRAN =        3387
$END
GRIDX(I) (I=1, 8):
-1000.0    -500.0    -100.0     300.0     500.0    1000.0    1500.0    2000.0
GRIDY(J) (J=1,12):
-2000.0   -1500.0   -1200.0    -900.0    -500.0    -200.0     200.0     500.0     900.0    1200.0
 1500.0    2000.0
PH(I,J) (I=1, 7),(J=1,11):
  .0000     .0000     .0000     .1000     .1000     .1000     .0000     .0000     .0000     .3000
  .3000     .2000     .2000     .1000     .0000     .1000     .4000     .5000     .5000     .2000
  .1000     .1000     .3000     .5000     .6000     .8000     .4000     .2000     .2000     .5000
  .6000     .8000     .9000     .7000     .3000     .4000     .6000     .5000     .6000     .8000
  .6000     .2000     .2000     .5000     .6000     .8000     .9000     .7000     .3000     .1000
  .3000     .5000     .6000     .8000     .4000     .2000     .0000     .1000     .4000     .5000
  .5000     .2000     .1000     .0000     .0000     .3000     .3000     .2000     .2000     .1000
  .0000     .0000     .0000     .1000     .1000     .1000     .0000
```

```
CUMULATIVE HIT PROBABILITY FOR ALL RUNS THIS SET:  PCH =   .3965

INPUT DATA FOR THE SET FOLLOW:
 $NL2
NRUN =       400C,NSHOT =          1,PTH = .50000000+000,TS1 = .10000000+002,TS2 = .25000000+002,CMAX = .90000000+002,
CMIN = .60000000+002,TR = .15000000+003,TCL = .16800000+004,TR2 = .38700000+005,TB = .12000000+003,APR = .75000000+003,
APA = .30000000+002,SIGTP = .20000000+001,SIGTC = .10000000+001,SIGTS = .50000000+000,SIGL = .10000000+003,SIGD = .50000000+003,
NX =          8,NY =          12,NRAN =      11482
$END
GRIDX(I) (I=1, 8):
-1000.0    -500.0    -100.0     300.0     500.0    1000.0    1500.0    2000.0
GRIDY(J) (J=1,12):
-2000.0   -1500.0   -1200.0    -900.0    -500.0    -200.0     200.0     500.0     900.0    1200.0
 1500.0    2000.0
PH(I,J) (I=1, 7),(J=1,11):
  .0000     .0000     .0000     .1000     .1000     .1000     .0000     .0000     .0000     .3000
  .3000     .2000     .2000     .1000     .0000     .1000     .4000     .5000     .5000     .2000
  .1000     .1000     .3000     .5000     .6000     .8000     .4000     .2000     .2000     .5000
  .6000     .8000     .9000     .7000     .3000     .4000     .6000     .5000     .6000     .8000
  .6000     .2000     .2000     .5000     .6000     .8000     .9000     .7000     .3000     .1000
  .3000     .5000     .6000     .8000     .4000     .2000     .0000     .1000     .4000     .5000
  .5000     .2000     .1000     .0000     .0000     .3000     .3000     .2000     .2000     .1000
  .0000     .0000     .0000     .1000     .1000     .1000     .0000
```

```
CUMULATIVE HIT PROBABILITY FOR ALL RUNS THIS SET:  PCH =   .3943

INPUT DATA FOR THE SET FOLLOW:
 $NL2
NRUN =       4000,NSHOT =          1,PTM = .50000000+000,TS1 = .10000000+002,TS2 = .25000000+002,CMAX = .90000000+002,
CMIN = .60000000+002,TR = .15000000+003,TCL = .16800000+004,TR2 = .38700000+005,TB = .12000000+003,APR = .75000000+003,
APA = .30000000+002,SIGTP = .20000000+001,SIGTC = .10000000+001,SIGTS = .50000000+000,SIGL = .10000000+003,SIGD = .50000000+003,
NX =          8,NY =          12,NRAN =       3382
$END
GRIDX(I) (I=1, 8):
-1000.0    -500.0    -100.0     300.0     500.0    1000.0    1500.0    2000.0
GRIDY(J) (J=1,12):
-2000.0   -1500.0   -1200.0    -900.0    -500.0    -200.0     200.0     500.0     900.0    1200.0
 1500.0    2000.0
PH(I,J) (I=1, 7),(J=1,11):
  .0000     .0000     .0000     .1000     .1000     .1000     .0000     .0000     .0000     .3000
  .3000     .2000     .2000     .1000     .0000     .1000     .4000     .5000     .5000     .2000
  .1000     .1000     .3000     .5000     .6000     .8000     .4000     .2000     .2000     .5000
  .6000     .8000     .9000     .7000     .3000     .4000     .6000     .5000     .6000     .8000
  .6000     .2000     .2000     .5000     .6000     .8000     .9000     .7000     .3000     .1000
  .3000     .5000     .6000     .8000     .4000     .2000     .0000     .1000     .4000     .5000
  .5000     .7000     .1000     .0000     .0000     .3000     .3000     .2000     .2000     .1000
  .0000     .0000     .0000     .1000     .1000     .1000     .0000

@FIN
```

APPENDIX F: INTERNAL VARIABLE AND FUNCTION NAMES

ACOSH(X): Statement function which computes the inverse hyperbolic cosine of X.

ACOTH(X): Statement function which computes the inverse hyperbolic cotangent of X.

APAI: Input variable APA converted to radians.

ASINH(X): Statement function which computes the inverse hyperbolic sine of X.

ATANH(X): Statement function which computes the inverse hyperbolic tangent of X.

CCM: Target course change magnitude (radians)

CCS: Indicates the direction of the target course change. CCM = +1 for a negative change (positive y direction or left turn). CCM = -1 for a positive change (negative y direction or right turn).

CR: Reciprocal of hydrodynamic coefficient c (yards) [see equation (B-14)]

DUM1, DUM2, DUM3, DUM4: Dummy variables for temporary storage of intermediate quantities

FNRUN: Floating point conversion of input variable NRUN

FNU0: ν_o-ratio of initial to terminal speed [see equation (B-20)]

NRAN1:	Running seed integer for random number subroutines UNFRM and GAUSS.
PCH:	Cumulative hit probability for set.
PCHS:	Cumulative hit probability for run.
RN:	Random number from a uniform distribution over the interval [0,1]
S:	arc-length of a turn maneuver (yards)
SIGTCI:	Input variable SIGTC converted to radians
SIGTSI:	Input variable SIGTS converted to yards per second
T1:	Actual time to weapon delivery (seconds)
T2:	Time required to complete a turn maneuver (seconds)
TAU:	Time variable Kt [see equations (B-19) and (B-27)]
TCLI:	Input variable TCL converted to yards
TCWD:	Target course at weapon delivery time (radians)
TM:	Time target maneuver starts (seconds)
TS1I:	Input variable TS1 converted to yards per second
TS2I:	Input variable TS2 converted to yards per second
TSE:	Target speed at end of turn (yards per second)

TST:	Terminal speed for maneuver (yards per second) [see equation (B-10)]
XAP:	X-COORDINATE OF AIMPOINT (YARDS)
XT:	x-coordinate of target (yards)
XTC:	Estimated x-coordinate of target (yards)
XWD:	x-coordinate of weapon delivery point (yards)
XWDG:	x-coordinate of weapon delivery point in hit probability grid coordinates (yards)
YAP:	y-coordinate of aimpoint (yards)
YT:	y-coordinate of target (yards)
YTC:	Estimated y-coordinate of target (yards)
YWD:	y-coordinate of weapon delivery point (yards)
YWDG:	y-coordinate of weapon delivery point in hit probability grid coordinates (yards)